REMARKS

The Office Action mailed October 12, 2010, considered and rejected claims 1-4, 7, 13 and 16-18-4, 7, 13 and 16-18. Claims 1-4, 7, 13 and 16-18 were rejected under 35 U.S.C. § 103(a) as being unpatentable over *White et al.* (Anatomy of a Commercial-Grade Immune System) in view of *Schultz et al.* (U.S. Publ. No. 2003/0065926) in further view of *Muttik et al.* (U.S. Publ. No. 2004/0199827).

The invention is directed generally to embodiments which record only interesting function calls that a code module makes ***during execution***. This is described primarily on page 7 of the specification. In particular, it states that "interesting behaviors are those which a user or implementer of the malware detection system has identified as interesting, potentially associated with malware, and are used to compare the behaviors of the code module against known malware behaviors." As an example, the table of page 7 lists many different function calls that could be considered as "interesting function calls." Once the interesting function calls have been recorded, they form the behavior signature for the code module. This behavior signature is compared to the behavior signature for known malware to determine whether the code module is malware. This comparison involves comparing the interesting function calls made by the code module ***during execution*** to those that are known to be made by known malware.

The current office action acknowledges that the previously cited art did not disclose creating a behavior signature from interesting function calls that a program makes during execution. However, to reject this feature, the current office action newly cites the Muttik reference.

Muttik discloses that external call characteristics of a program can be determined and compared to those of known malware programs. However, Muttik determines what external calls a program makes, not by executing the program, but by directly reading the instructions of the compiled code including the import table. For example, paragraph 40 states that an embedded runtime library can be searched for characteristic API external calls that are made from within the runtime library. The other portions of the program can also be read to locate the external calls. Muttik therefore does not perform malware detection by executing the program. As such, Muttik cannot disclose recording "interesting function calls that the code module makes ***as it is executed***."

Schultz, similarly, is related to virus detection in binaries that is performed without executing the binaries. *See* ¶ 42. Because the binaries are not executed, Schultz is not relevant to behavioral based detection because it requires execution to detect which function calls are made. Further, White only tangentially mentions that virus detection can be performed by simulating its behavior. However, nothing more is stated about what this simulation would entail. As such, there is nothing in White that would teach or suggest that only interesting function calls that are made during execution can be recorded to define a behavior signature.

Finally, it is noted that it cannot be simply argued that Muttik could be modified to perform its detection by monitoring the program during execution. <u>***Muttik performs static signature detection***</u>. Static signature comparison has its limits as is discussed in the background of the present invention.

In contrast, the invention performs dynamic malware detection. A program's behavior is dynamic. The invention provides the benefit, among others, of detecting unknown viruses based on the dynamic behavior of a program as it is executed that may indicate the presence of malware. Muttik's approach, because it is static, cannot be used in this manner. For example, Muttik would only detect malware if the malware's static signature were known and the program contained the same static signature. In contrast, the present invention provides much better malware detection capabilities because it accounts for malware which is not identical to known malware, but is similar enough to be detected by its behavior. *See* Pg. 3, lines 17-24. As such, the present invention is designed to detect new malware that is derived from known malware, but is different enough that it would not likely be detected by static signature comparison (whether the static signature was made of only external calls as in Muttik or otherwise).

In summary, none of the cited references disclose or suggest that only interesting function calls are recorded <u>***during executing***</u> of a program, or that these interesting function calls can be used to determine whether the potential malware is in fact malware by comparing the interesting function calls to those made by known malware <u>***during execution***</u>. As such, these references in combination fail to teach or suggest:

> at least one dynamic behavior evaluation module, wherein each dynamic behavior evaluation module provides a virtual environment for executing a code module of a particular type, and <u>*wherein each dynamic behavior evaluation module records interesting function calls that the code module makes **as it is executed**, wherein the interesting function calls are specified by a user and comprise a subset of all function calls that the code module makes, wherein only*</u>

> *the interesting function calls, but not all function calls, that the code module makes* ***during execution*** *in the dynamic behavior evaluation module are recorded into a behavior signature corresponding to the code module*;
>
> a management module, wherein the management module obtains the code module, and wherein the management module evaluates the code module to determine the code module's type, and wherein the management module selects a dynamic behavior evaluation module to execute the code module according to the code module's type;
>
> a malware behavior signature store storing at least one known malware ***behavior*** signature of a known malware, *wherein each of the at least one known malware behavior signature is comprised of only interesting function calls as specified by the user*;
>
> a behavior signature comparison module that obtains the behavior signature of the code module and compares the behavior signature of the code module to the known malware behavior signatures in the malware behavior signature store *to determine whether the interesting function calls recorded in the behavior signature of the code module match the interesting function calls in any of the known malware behavior signatures*; and
>
> wherein the malware detection system is configured to report whether the code module is a known malware based at least in part on the degree that *the interesting function calls recorded in the behavior signature of the code module match the interesting function calls in a behavior signature of the known malware*;

as claimed in claim 1, or as similarly claimed in the remaining independent claims. Applicant, therefore, respectfully requests that the rejections be withdrawn.

In view of the foregoing, Applicant respectfully submits that the other rejections to the claims are now moot and do not, therefore, need to be addressed individually at this time. In the event that the Examiner finds remaining impediment to a prompt allowance of this application that may be clarified through a telephone interview, the Examiner is requested to contact the undersigned attorney at (801) 322-8427.

The Commissioner is hereby authorized to charge payment of any of the following fees that may be applicable to this communication, or credit any overpayment, to Deposit Account No. 23-3178: (1) any filing fees required under 37 CFR § 1.16; and/or (2) any patent application and reexamination processing fees under 37 CFR § 1.17; and/or (3) any post issuance fees under 37 CFR § 1.20. In addition, if any additional extension of time is required, which has not otherwise been requested, please consider this a petition therefore and charge any additional fees that may be required to Deposit Account No. 23-3178.

Dated this 2nd day of November, 2010.

Respectfully submitted,

/BRIAN D. TUCKER/

RICK D. NYDEGGER
Registration No. 28,651
BRIAN D. TUCKER
Registration No. 61,550
Attorneys for Applicant
Customer No. 47973

RDN:BDT:cj
3084536_1